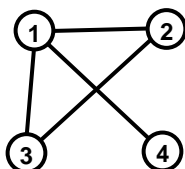


Terminologie Grafuri

- **Graf** = o mulțime finită V care are asociată o relație binară internă L . Graful se va nota $G = (V, L)$.
 G – graful, V – mulțimea vârfurilor, L – mulțimea legăturilor între vârfuri.

Exemplu: $G = (V, L)$, $V = \{1, 2, 3, 4\}$ $L = \{(1,2), (1,3), (1,4), (2,3), (2,4)\}$

- **Graf neorientat** = un graf în care mulțimea legăturilor L este **simetrică** – dacă elementul (p, q) aparține L atunci și elementul (q, p) aparține L .

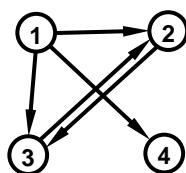


$G_1 = (V, L)$

$V = \{1, 2, 3, 4\}$

$L = \{(1,2), (2,1), (1,3), (3,1), (1,4), (4,1), (2,3), (3,2)\}$

- **Graf orientat** = un graf în care mulțimea legăturilor L nu este simetrică.



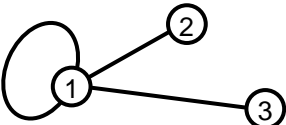
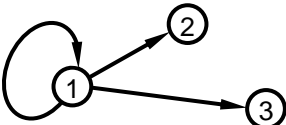
$G_2 = (V, L)$

$V = \{1, 2, 3, 4\}$

$L = \{(1,2), (1,3), (1,4), (2,3), (2,4)\}$

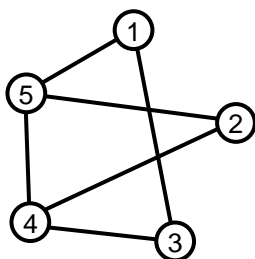
În grafuri neorientate	În grafuri orientate
• Nod sau vârf = element al mulțimii V	• Vârf = element al mulțimii V
• Muchie = element al mulțimii L (legătură între două noduri). Se poate parcurge în ambele sensuri.	• Arc = element al mulțimii L (legătură între două vârfuri). Se poate parcurge numai în sensul indicat de săgeată.
• Adiacență = două noduri sunt adiacente dacă există muchie între ele. • (sau) Două noduri p și q sunt adiacente dacă elementul (p,q) aparține mulțimii L . Exemplu: nodurile 1 și 2 din graful G_1 de mai sus sunt adiacente. Nodurile 2 și 4 nu sunt. Observație: dacă nodul p este adiacent cu nodul q atunci și nodul q este adiacent cu nodul p (valabil numai în grafuri neorientate).	• Adiacență = două vârfuri p și q sunt adiacente dacă există arc de la p la q . • (sau) Două vârfuri p și q sunt adiacente dacă elementul (p,q) aparține mulțimii L . Exemplu: vârful 1 este adiacent cu vârful 2 în graful G_2 de mai sus. Vârful 2 nu este adiacent cu vârful 1.
• Incidență = o muchie este incidentă cu un nod dacă îl are ca extremitate. Exemplu: în graful G_1 muchia $(1,2)$ este incidentă cu nodurile 1 și 2.	• Incidență spre interior = un arc (p,q) este incident spre interior cu vârful q (vârful spre care merge). • Incidență spre exterior = un arc (p,q) este incident spre exterior cu vârful p (vârful din care pornește). Exemplu: în graful G_2 arcul $(1,2)$ este incident spre interior cu vârful 2 și incident spre exterior cu vârful 1.

În grafuri neorientate	În grafuri orientate
<ul style="list-style-type: none"> • Grad = numărul de noduri incidente cu un nod. • (sau) Numărul de muchii incidente cu un nod. <p>Exemplu: în G_1 nodul 1 are gradul 3 iar nodul 4 are gradul 1.</p> <p>Observație: pentru gradul nodului i se folosește notația: d_i (d de la „degree” – grad în engleză)</p>	<ul style="list-style-type: none"> • Grad = numărul de arce incidente spre interior plus numărul de arce incidente spre exterior cu un vârf (se notează cu d_i unde i este numărul vârfului). • Grad interior = numărul de arce incidente spre interior (se notează cu d_i^+). • Grad exterior = numărul de arce incidente spre exterior (se notează cu d_i^-). <p>Observație: pentru un vârf i avem $d_i = d_i^+ + d_i^-$.</p> <p>Exemplu: în G_2 vârful 2 are gradul 3, gradul interior 2 și gradul exterior 1.</p>
<ul style="list-style-type: none"> • Nod izolat = nod cu gradul 0. • (sau) Nod de care nu este legată nici o muchie. 	<ul style="list-style-type: none"> • Vârf izolat = vârf cu gradul 0. • Intrare = vârf cu gradul interior 0. • (sau) Vârf în care nu intră nici un arc. <p>Exemplu: în G_2 vârful 1 este intrare.</p> <ul style="list-style-type: none"> • Ieșire = vârf cu gradul exterior 0. • (sau) Vârf din care nu pleacă nici un arc. <p>Exemplu: în G_2 vârful 4 este ieșire.</p>
<ul style="list-style-type: none"> • Nod terminal = nod cu gradul 1. <p>Exemplu: în G_1 nodul 4 este terminal.</p>	<ul style="list-style-type: none"> • Vârf terminal = vârf cu gradul 1.
<ul style="list-style-type: none"> • Lanț = șir de noduri cu proprietatea că oricare două noduri consecutive sunt adiacente. <p>Exemplu: în G_1 $\{1, 2, 3\}$ și $\{1, 3, 2, 1, 4\}$ sunt lanțuri.</p>	<ul style="list-style-type: none"> • Drum = șir de vârfuri cu proprietatea că oricare două vârfuri consecutive sunt adiacente. <p>Exemplu: în G_2 $\{1, 2, 3\}$ este drum.</p>
<ul style="list-style-type: none"> • Lungime lanț = numărul de muchii din care este format (numărul de noduri minus 1). <p>Exemplu: în G_1 lungimea lanțului $\{1, 2, 3\}$ este 2 și lungimea lanțului $\{1, 3, 2, 1, 4\}$ este 4.</p>	<ul style="list-style-type: none"> • Lungime drum = numărul de arce din care este format (numărul de vârfuri minus 1).
<ul style="list-style-type: none"> • Lanț simplu = lanț în care nu se repetă muchii. <p>Exemplu: în G_1 $\{1, 2, 3\}$ și $\{1, 3, 2, 1, 4\}$ sunt lanțuri simple.</p>	<ul style="list-style-type: none"> • Drum simplu = drum în care nu se repetă arce. <p>Exemplu: în G_2 $\{1, 2, 3\}$ și $\{1, 3, 2\}$ sunt drumuri simple.</p>
<ul style="list-style-type: none"> • Lanț compus = lanț în care se repetă muchii. 	<ul style="list-style-type: none"> • Drum compus = drum în care se repetă arce.
<ul style="list-style-type: none"> • Lanț elementar = lanț în care nu se repetă noduri. 	<ul style="list-style-type: none"> • Drum elementar = drum în care nu se repetă vârfuri.
<ul style="list-style-type: none"> • Lanț neelementar = lanț în care se repetă noduri. <p>Observație: un lanț elementar este și simplu.</p> <p>Observație: un lanț compus este și neelementar.</p>	<ul style="list-style-type: none"> • Drum neelementar = drum în care se repetă vârfuri. <p>Observație: un drum elementar este și simplu.</p> <p>Observație: un drum compus este și neelementar.</p>

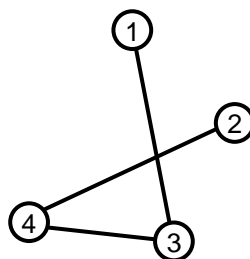
În grafuri neorientate	În grafuri orientate
<ul style="list-style-type: none"> • Ciclu = lanț în care primul nod coincide cu ultimul. Este elementar dacă nodurile nu se repetă (cu excepția primului și ultimului). 	<ul style="list-style-type: none"> • Circuit = drum în care primul vârf coincide cu ultimul. Este elementar dacă vârfurile nu se repetă (cu excepția primului și ultimului).
<ul style="list-style-type: none"> • Bucla = ciclu format dintr-o singură muchie (se întâlnește rar). <p>Exemplu:</p> 	<ul style="list-style-type: none"> • Bucla = circuit format dintr-un singur arc (se întâlnește rar). <p>Exemplu:</p> 

- **Subgraf** = graf care se obține din alt graf eliminând noduri / vârfuri și toate muchiile / arcele incidente cu acestea.

Exemplu:



$G_3 = (V_3, L_3)$



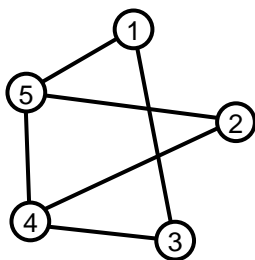
$G'_3 = (V'_3, L'_3)$ subgraf al lui G_3

Observație: V'_3 este submulțime a lui V_3 și L'_3 este submulțime a lui L_3 .

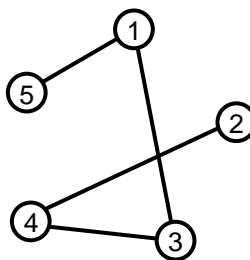
Observație: numărul de subgrafuri ale unui graf dat este $2^n - 1$, unde n este numărul de noduri

- **Graf parțial** = graf care se obține din alt graf eliminând muchii / arce.

Exemplu:



$G_4 = (V_4, L_4)$

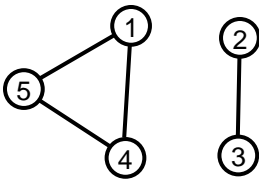
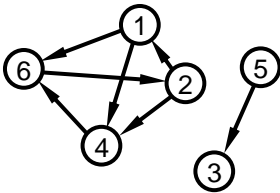


$G'_4 = (V_4, L'_4)$ graf parțial al lui G_4

Observație: L'_4 este submulțime a lui L_4 .

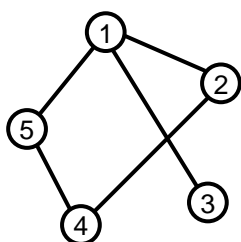
Observație: numărul de grafuri parțiale ale unui graf dat este 2^m , unde m este numărul de muchii

- **Graf regulat** = graf neorientat în care toate nodurile au același grad.

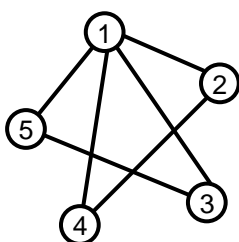
În grafuri neorientate	În grafuri orientate
<ul style="list-style-type: none"> • Graf complet = graf neorientat în care există muchie între oricare două noduri. <p>Observație: numărul de muchii într-un graf complet se calculează cu formula:</p> $\frac{n \cdot (n - 1)}{2}$ <p>unde n este numărul de noduri.</p>	<ul style="list-style-type: none"> • Graf turnir = graf orientat în care între oricare două noduri există un singur arc.
<ul style="list-style-type: none"> • Graf conex = graf în care între oricare două noduri există cel puțin un lanț care le unește. 	<ul style="list-style-type: none"> • Graf tare conex = graf în care între oricare două vârfuri există cel puțin un drum care le unește.
<ul style="list-style-type: none"> • Componentă conexă = subgraf al grafului de referință maximal în raport cu proprietatea de conexitate. • (sau) Grup de noduri astfel încât există lanț între oricare două noduri din grup (grupul este conex) dar nu există muchii de la nodurile din grup la celelalte noduri ale grafului (grupul este „izolat” de restul grafului). <p>Exemplu:</p>  <p>Graful are două componente conexe {1, 4, 5} și {2, 3}</p>	<ul style="list-style-type: none"> • Componentă tare conexă = subgraf al grafului de referință maximal în raport cu proprietatea de tare conexitate. • (sau) Grup de vârfuri astfel încât există drum între oricare două vârfuri din grup (grupul este conex) dar nu există arce de la vârfurile din grup la celelalte vârfuri ale grafului (grupul este „izolat” de restul grafului). <p>Exemplu:</p>  <p>Graful are trei componente tari conexe {1, 2, 4, 6}, {3}, {5}</p>
<ul style="list-style-type: none"> • Lanț hamiltonian = lanț elementar care conține toate nodurile unui graf. 	<ul style="list-style-type: none"> • Drum hamiltonian = drum elementar care conține toate vârfurile unui graf.
<ul style="list-style-type: none"> • Ciclu hamiltonian = ciclu elementar care conține toate nodurile unui graf. 	<ul style="list-style-type: none"> • Circuit hamiltonian = circuit elementar care conține toate vârfurile unui graf.
<ul style="list-style-type: none"> • Graf hamiltonian = graf care conține un ciclu hamiltonian. <p>Condiție suficientă: dacă gradul oricărui nod este mai mare decât $n/2$ și sunt cel puțin 3 noduri atunci graful este hamiltonian.</p> <p>Observație: condiția de mai sus este doar suficientă nu și necesară adică este posibil să existe grafuri hamiltoniene care nu respectă condiția. (Dacă respectă condiția atunci este hamiltonian, dacă nu atunci poate este, poate nu).</p>	<ul style="list-style-type: none"> • Graf hamiltonian = graf care conține un circuit hamiltonian.
<ul style="list-style-type: none"> • Lanț eulerian = lanț simplu care conține toate muchiile unui graf. 	<ul style="list-style-type: none"> • Drum eulerian = drum simplu care conține toate arcele unui graf.
<ul style="list-style-type: none"> • Ciclu eulerian = ciclu simplu care conține toate muchiile unui graf. 	<ul style="list-style-type: none"> • Circuit eulerian = circuit simplu care conține toate arcele unui graf.

În grafuri neorientate	În grafuri orientate
<ul style="list-style-type: none"> • Graf eulerian = graf care conține un ciclu eulerian. <p>Teoremă: Un graf neorientat este eulerian dacă și numai dacă gradul oricărui nod este par.</p>	<ul style="list-style-type: none"> • Graf eulerian = graf care conține un circuit eulerian. <p>Teoremă: Un graf orientat este eulerian dacă și numai dacă pentru orice vârf gradul interior este egal cu gradul exterior.</p>

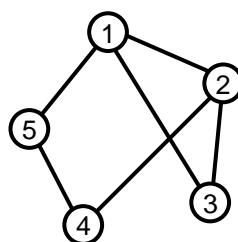
Exemple:



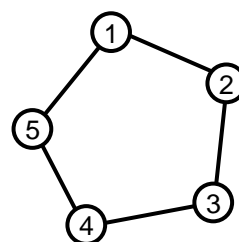
nici hamiltonian
nici eulerian



nu este hamiltonian
este eulerian



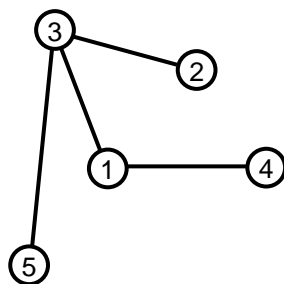
este hamiltonian
nu este eulerian



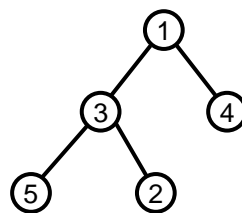
și hamiltonian
și eulerian

Arbori

- **Arbore** = graf conex și fără cicluri.
- **(sau)** Graf fără cicluri în care $m = n - 1$, unde m = numărul de muchii; n = numărul de noduri.
- **(sau)** Graf conex în care $m = n - 1$, unde m = numărul de muchii; n = numărul de noduri.
- **(sau)** Graf în care orice pereche de noduri este legată de un lanț și numai unul.
- **(sau)** Graf fără cicluri, maximal (orice muchie adăugată în plus formează un ciclu).
- **(sau)** Graf conex, minimal (dacă se șterge vreuna din muchiile existente se pierde conexitatea).
- **Rădăcină** = nod special care generează așezarea nodurilor pe nivele.
- **(sau – definiție recursivă arbore)** Un arbore este format dintr-o rădăcină la care sunt conectați un număr finit de arbori.
- **Descendent indirect (urmaș)** = într-un arbore cu rădăcină un nod x este descendentul (urmașul) unui nod y dacă există un lanț care le unește și nu trece prin rădăcină și x se află pe un nivel mai mare decât y (sub y).
- **Descendent direct (fiu)** = într-un arbore cu rădăcină un nod x este descendentul direct (fiul) unui nod y dacă există muchie între ele și nivelul lui x este mai mare decât al lui y .
- **Ascendent indirect (strămoș)** = într-un arbore cu rădăcină un nod x este ascendentul (strămoșul) unui nod y dacă există un lanț care le unește și nu trece prin rădăcină și x se află pe un nivel mai mic decât y (deasupra lui y).
- **Ascendent direct (tată)** = într-un arbore cu rădăcină un nod x este ascendentul direct (tatăl) unui nod y dacă există muchie între ele și nivelul lui x este mai mic decât al lui y .
- **Frați** = noduri care au același tată.
- **Frunze (noduri terminale)** = nodurile care nu au descendenți.
- **Nivelul unui nod** într-un arbore cu rădăcină = lungimea lanțului de la acesta la rădăcină. Rădăcina se află pe nivelul 0.

Exemple:

arbore fără rădăcină



arbore cu rădăcină

1 = rădăcină
 2,3,4,5 = descendenții lui 1
 2,5 = fii lui 3
 1,3 = ascendenții lui 2
 3 = tatăl lui 2
 3, 4 = frați
 2,4,5 = frunze
 1 = nivelul 0
 3,4 = nivelul 1
 2,5 = nivelul 2

• **Înălțimea unui arbore:** lungimea celui mai lung lanț de la rădăcină la o frunză; nivelul maxim.

• **Arbore multicai** = arbore cu rădăcină în care nodurile pot avea mai mult de doi fii.

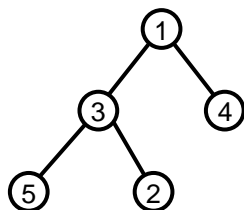
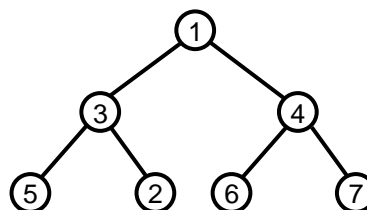
• **Arbore binar** = arbore cu rădăcină în care orice nod are cel mult doi fii.

Observație: într-un arbore binar, dacă nivelul k este complet (conține toate nodurile posibile) atunci pe nivelul k vor fi 2^k noduri.

• **Arbore binar complet** = arbore cu rădăcină în care toate nivele sunt complete (au 2^k noduri, unde k este numărul nivelului).

Observație: într-un arbore binar complet cu n nivele sunt $2^{n+1}-1$ noduri ($2^0 + 2^1 + 2^2 + \dots + 2^n$).

• **Arbore binar aproape complet** = arbore binar care are toate nivelele complete cu excepția ultimului care este incomplet. Nodurile de pe ultimul nivel vor fi grupate la stânga sau la dreapta. Exemple:

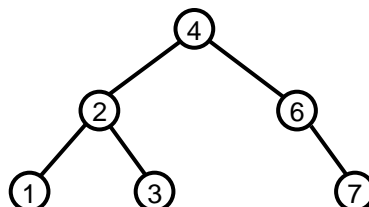
Arbore binar
aproape complet

Arbore binar complet

• **Arbore binar echilibrat** = arbore binar în care, pentru orice nod, înălțimea arborelui format de descendenții din stânga diferă cu cel mult 1 de înălțimea arborelui format de descendenții din dreapta. Graful din stânga în figura de mai sus este un exemplu de astfel de arbore.

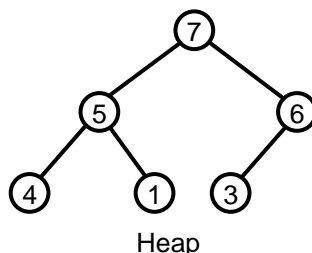
• **Arbore binar perfect echilibrat** = arbore binar în care, pentru orice nod, înălțimea arborelui format de descendenții din stânga este egală cu înălțimea arborelui format de descendenții din dreapta. Graful din dreapta în figura de mai sus este un exemplu de astfel de arbore.

• **Arbore de căutare** = arbore binar în care, pentru orice nod, descendenții din stânga sunt mai mici și cei din dreapta sunt mai mari decât el. Exemplu:



Arbore binar de căutare

- **Heap** = arbore binar aproape complet în care fiecare pereche tată – fiu respectă o condiție dată. De exemplu orice tată este mai mare decât fii lui:



- **Arbore parțial** = arbore care este graf parțial al unui graf dat.

Toate noțiunile de mai sus referitoare la arbori sunt valabile pentru grafuri neorientate. În cazul grafurilor orientate se mai adaugă noțiunea de **arborescență**:

- **Arborescență** = graf orientat care este arbore și admite rădăcină. Un graf orientat admite vârful x ca rădăcină dacă pentru oricare alt vârf y al grafului există un drum care merge de la x la y (există drumuri de la x la toate celelalte vârfuri).

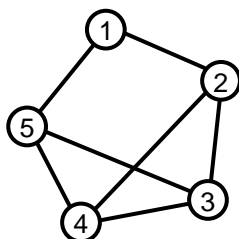
Observație: dacă un arbore orientat admite rădăcină atunci aceasta este unică. [dacă ar admite două rădăcini x și y atunci există drum de la x la y și de la y la x . Cele două drumuri formează un circuit deci graful nu este arbore – graful trebuie să fie conex și **fără cicluri**]

Metode de reprezentare grafuri

- **Matrice de adiacență**

Este o matrice în care elementul de pe linia i și coloana j este 1 dacă există legătură între nodurile / vârfurile i și j .

$MA[i][j] = 1$ dacă (i, j) aparține mulțimii L , $G = (V, L)$.



$$MA = \begin{bmatrix} 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 \\ 0 & 1 & 0 & 0 & 1 \\ 0 & 1 & 0 & 0 & 1 \\ 1 & 0 & 1 & 1 & 0 \end{bmatrix}$$

Observație 1: dacă graful este neorientat matricea este simetrică față de diagonala principală.

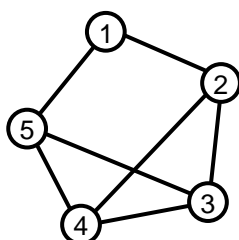
Observație 2: diagonala principală conține numai valori de 0.

Observație 3: pentru grafuri neorientate numărul de valori de 1 de pe linia sau coloana i reprezintă gradul nodului i .

Observație 4: pentru grafuri orientate numărul de valori de 1 de pe linia i reprezintă gradul exterior al vârfului i (numărul de arce care ies). Numărul de valori de 1 de pe coloana i reprezintă gradul interior al vârfului i (numărul de arce care intră).

- **Liste de adiacență**

Listele de adiacență rețin pentru fiecare nod vecinii lui (nodurile adiacente cu el).



1: 2, 5
 2: 1, 3, 4
 3: 2, 4, 5
 4: 2, 3, 5
 5: 1, 3, 4

Metode de reprezentare arbori

- **Matrice de adiacență**

Se procedează exact ca la un graf obișnuit.

- **Liste de descendenți**

Se procedează la fel ca la listele de adiacență prezentate mai sus dar în listă vor apare numai descendenții direcți.

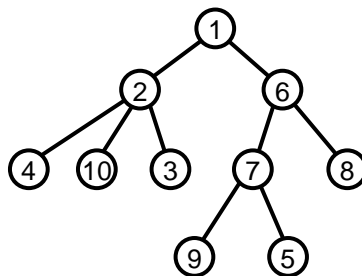
- **Vectorul taților**

Într-un tablou, pe poziția corespunzătoare fiecărui nod se memorează părintele nodului respectiv. Pe poziția corespunzătoare rădăcinii se memorează valoarea 0.

- **Memorare dinamică (arbori binari)**

Pentru arborii binari există și posibilitatea de a memora arborele sub forma unei structuri înlanțuite. Fiecare element al structurii va avea trei câmpuri: un câmp pentru informația utilă din nod, un câmp pentru adresa fiului din stânga și un câmp pentru adresa fiului din dreapta.

Exemple:



$$\begin{bmatrix}
 0 & 1 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
 1 & 0 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 1 \\
 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\
 1 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 \\
 0 & 0 & 0 & 0 & 1 & 1 & 0 & 0 & 1 & 0 \\
 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\
 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\
 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0
 \end{bmatrix}$$

Matricea de adiacență

1: 2, 6
 2: 3, 4, 10
 3:
 4:
 5:
 6: 7, 8
 7: 5, 9
 8:
 9:
 10:

Liste de descendenți

(0,1,2,2,7,1,6,6,7,2)

Vectorul taților